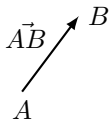


На пороге компьютерной графики.

Храмов Д. А.

28.12.2019

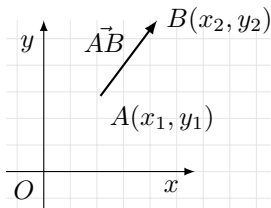
# Вектор



**Вектор** — математический объект, который характеризуется длиной и направлением.

**Вектор  $\vec{AB}$**  — упорядоченная пара точек  $A$  (начало вектора) и  $B$  (конец вектора).

## Система координат

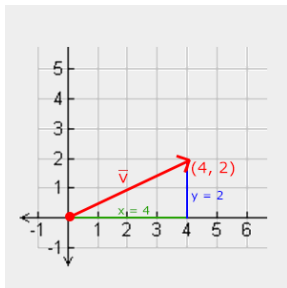


Дает возможность ввести координаты вектора:

$$\vec{AB} = (x_2 - x_1, y_2 - y_1).$$

**Внимание!** В программировании, для краткости, “вектором” называют одномерный массив чисел. Такой массив действительно может представлять координаты вектора. Но сам вектор — это геометрический объект, который от выбора системы координат не зависит.

## Длина вектора (норма)



$$||v|| = \sqrt{x^2 + y^2}$$

```
p = [4,2];  
norm(p)
```

```
ans =  
    4.4721
```

## Скалярное произведение векторов (dot product)

$$\vec{v} \cdot \vec{k} = ||\vec{v}|| \cdot ||\vec{k}|| \cdot \cos \theta$$

$$\begin{bmatrix} 0.6 & -0.8 & 0 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} = (0.6 \cdot 0) + (-0.8 \cdot 1) + (0 \cdot 0) = -0.8.$$

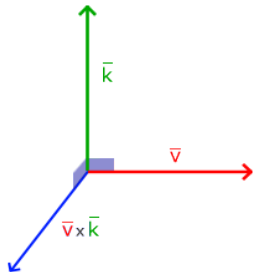
`v = [0.6,-0.8,0]; k = [0,1,0];`

`dot(v,k)`

`ans =`

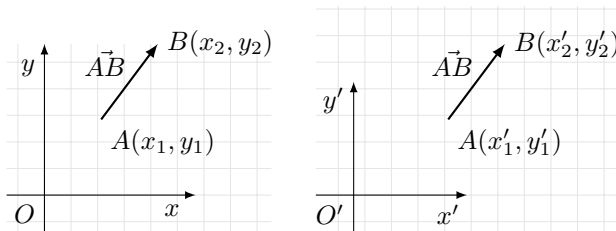
`-0.8000`

## Векторное произведение векторов (cross product)



`cross(v,k)`

# Координаты вектора в разных системах координат

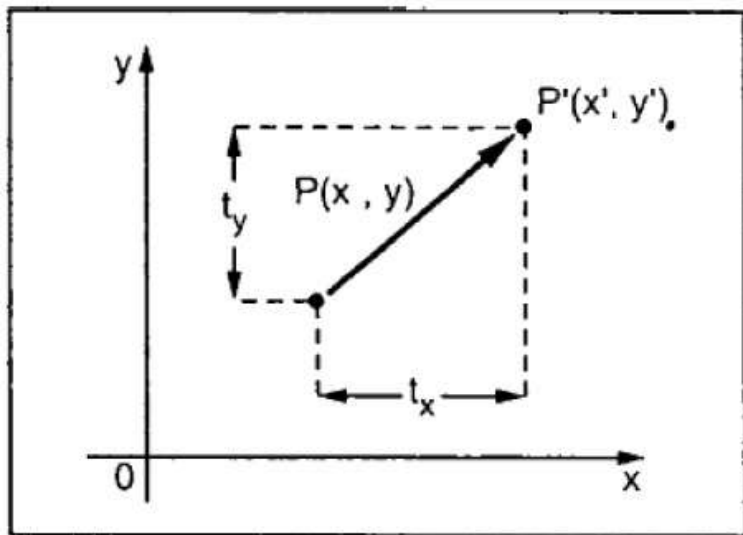


В различных системах координат один и тот же вектор имеет разные координаты.

Трактовки преобразования координат:

1. Координаты вектора остаются неизменными, а изменяется система координат.
2. Изменяются координаты вектора, а система координат остается неизменной.

## Перенос (трансляция, translation)





## Связь новых координат со старыми

$$\begin{aligned}x' &= x + t_x, \\ y' &= y + t_y.\end{aligned}$$

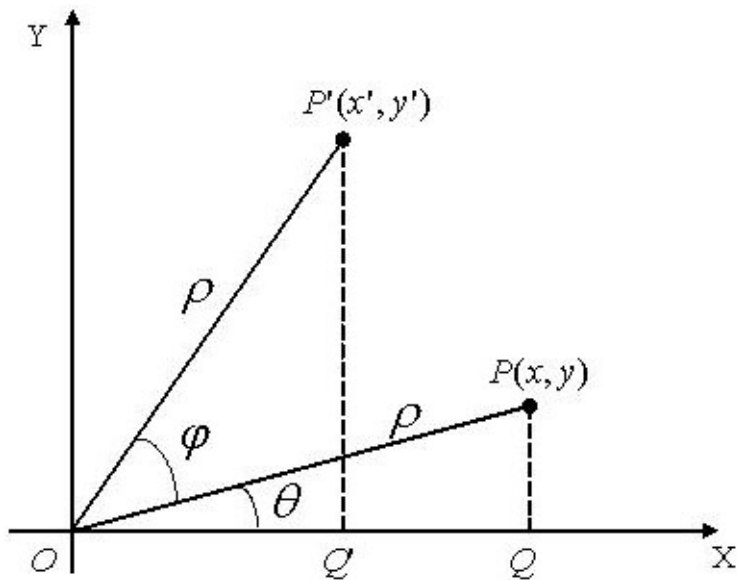
Введем матричные обозначения

$$P' = \begin{bmatrix} x' \\ y' \end{bmatrix}, \quad P = \begin{bmatrix} x \\ y \end{bmatrix}, \quad T = \begin{bmatrix} t_x \\ t_y \end{bmatrix}.$$

В матричной записи получим

$$P' = P + T.$$

## Поворот (rotation)



## Формулы для пересчета старых координат точки $P$ в новые $P'$

Пусть  $Q$  и  $Q'$  — проекции точек  $P(x, y)$  и  $P'(x', y')$  соответственно на ось  $X$ .

Тогда из прямоугольного треугольника  $OP'Q'$  имеем:

$$\begin{aligned}\cos(\theta + \varphi) &= \frac{x'}{\rho}, \\ \sin(\theta + \varphi) &= \frac{y'}{\rho}.\end{aligned} \Rightarrow \begin{aligned}x' &= \rho \cos(\theta + \varphi), \\ y' &= \rho \sin(\theta + \varphi).\end{aligned}$$

Раскрывая суммы  $\cos()$  и  $\sin()$ , получим

$$\begin{aligned}x' &= \rho \cos \theta \cos \varphi - \rho \sin \theta \sin \varphi, \\ y' &= \rho \sin \theta \cos \varphi + \rho \cos \theta \sin \varphi.\end{aligned} \Rightarrow \begin{aligned}x' &= x \cos \varphi - y \sin \varphi, \\ y' &= x \sin \varphi + y \cos \varphi,\end{aligned}$$

поскольку  $x = \rho \cos \theta$ ,  $y = \rho \sin \theta$ .

## Формулы преобразования координат в матричной записи

$$\begin{aligned}x' &= x \cos \varphi - y \sin \varphi, \\y' &= x \sin \varphi + y \cos \varphi,\end{aligned}$$

Вспомнив, что

$$P' = \begin{bmatrix} x' \\ y' \end{bmatrix}, \quad P = \begin{bmatrix} x \\ y \end{bmatrix},$$

запишем формулы преобразования координат в матричном виде

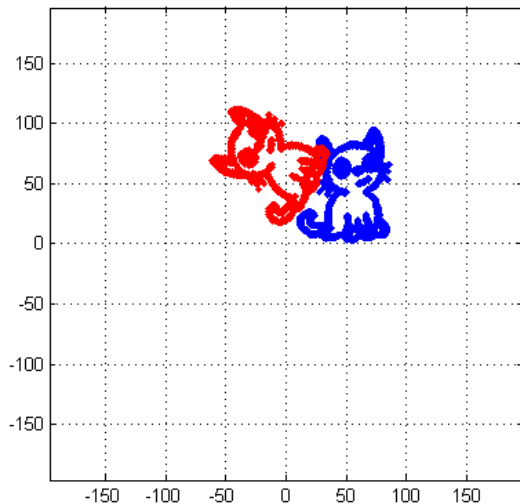
$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos \varphi & -\sin \varphi \\ \sin \varphi & \cos \varphi \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}.$$

$$P' = RP,$$

где  $R$  – матрица поворота:

$$R = \begin{bmatrix} \cos \varphi & -\sin \varphi \\ \sin \varphi & \cos \varphi \end{bmatrix}.$$

## Пример. Преобразование координат изображения



## Код примера

```
A = imread('kitten.bmp');  
[N,M] = size(A);  
[i,j] = find(~A);  
x = j; y = M-i;  
plot(x,y, '.')  
v = 2*max([N,M]);  
grid on  
axis([-v v -v v])  
axis square  
hold on  
  
R = @(phi) [cos(phi),sin(phi); -sin(phi),cos(phi)];  
  
a =-pi/3;  
B = R(a)*[x';y'];  
plot(B(1,:),B(2,:), 'r.')
```

## Матрица преобразования координат

Для произвольной матрицы  $A$  размерности  $2 \times 2$  можно записать аналогично

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}.$$

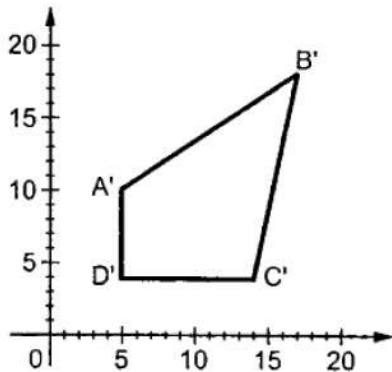
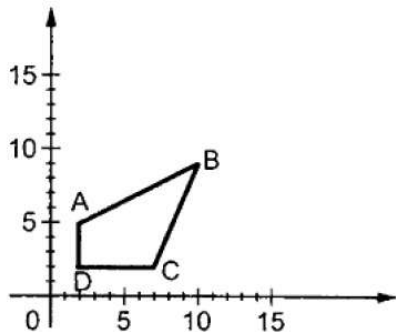
или

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = A \begin{bmatrix} x \\ y \end{bmatrix},$$

где  $A$  – матрица преобразования координат

$$A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}.$$

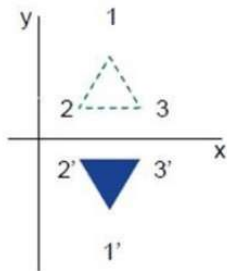
## Масштабирование (scaling)



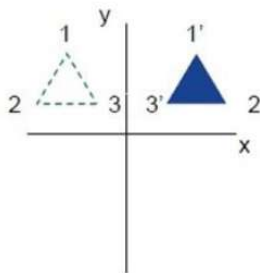
$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} S_x & 0 \\ 0 & S_y \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}.$$



## Отражение (reflection)



(a)



(b)

$$\text{a) } \begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \quad \text{b) } \begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}.$$

## Последовательные преобразования

Пусть координаты точки  $P(x, y)$  преобразуются сначала в координаты  $P'(x', y')$  при помощи матрицы  $B$ , а затем в координаты  $P''(x'', y'')$  при помощи матрицы  $A$

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = B \begin{bmatrix} x \\ y \end{bmatrix}, \quad \begin{bmatrix} x'' \\ y'' \end{bmatrix} = A \begin{bmatrix} x' \\ y' \end{bmatrix}.$$

Тогда матрица  $C$ , выполняющая преобразование от  $P(x, y)$  к  $P''(x'', y'')$ ,

$$\begin{bmatrix} x'' \\ y'' \end{bmatrix} = AB \begin{bmatrix} x \\ y \end{bmatrix} \Rightarrow \begin{bmatrix} x'' \\ y'' \end{bmatrix} = C \begin{bmatrix} x \\ y \end{bmatrix},$$

запишется в виде

$$C = AB$$

или

$$C = \begin{bmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{bmatrix} = \begin{bmatrix} a_{11}b_{11} + a_{12}b_{21} & a_{11}b_{12} + a_{12}b_{22} \\ a_{21}b_{11} + a_{22}b_{21} & a_{21}b_{12} + a_{22}b_{22} \end{bmatrix} = AB.$$

## Однородные координаты

Большинство операций по преобразованию координат выполняется умножением матриц, тогда как перенос – суммированием.

Для единообразия, в компьютерной графике вводят однородные координаты для плоскости

$$\begin{bmatrix} x \\ y \end{bmatrix} \Rightarrow \begin{bmatrix} x \\ y \\ 1 \end{bmatrix},$$

а вместо матриц размерности  $2 \times 2$  (для плоскости) используют матрицы размерности  $3 \times 3$ , что позволяет записать все преобразования, используя только умножение матриц. В том числе и перенос

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}.$$

## Некоммутативность умножения матриц

Выполним в разной последовательности масштабирование и перенос начала отсчета

$$S = \begin{bmatrix} S_x & 0 & 0 \\ 0 & S_y & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad T = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix}.$$

Получим

$$ST = \begin{bmatrix} S_x & 0 & S_x t_x \\ 0 & S_y & S_y t_y \\ 0 & 0 & 1 \end{bmatrix}, \quad TS = \begin{bmatrix} S_x & 0 & t_x \\ 0 & S_y & t_y \\ 0 & 0 & 1 \end{bmatrix}.$$

Тем не менее, произведение некоторых видов матриц коммутативно. Например, диагональные матрицы переставлять можно. Вспомним геометрический смысл осуществляемого ими преобразования.

## Обратное преобразование

Преобразование, обратное тому, что осуществляется матрицей  $A$  выполняется при помощи обратной к  $A$  матрицы  $A^{-1}$ .

Для прямой и обратной матриц справедливо

$$AA^{-1} = E,$$

где  $E$  – единичная матрица.

1. LearnOpenGL - Transformations – иллюстрации к норме и произведениям векторов.
2. TutorialPoint: 2D Transformation – картинки с переносом, масштабированием и отражением.
3. Representing Data as a Surface - MATLAB & Simulink – график поверхности, код примера построения графика `sinc`.
4. MATLAB Plot Gallery – просто красивые и полезные примеры.